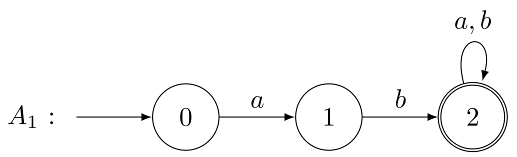
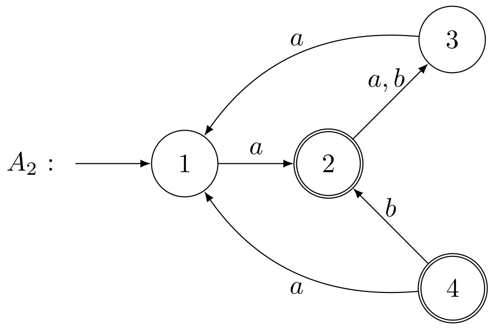
Fiche TD3 : Opérations sur les automates

Exercice 1 (*) *Lectures dans un automate*

Dans cet exercice, on considère les automates A_1 et A_2 ci-dessous :





Répondre aux questions suivantes :

1. L'automate A_1 est-il émondé ? Est-il déterministe ? Est-il complet ?

2. Dans quel état se trouve-t-on dans A_1 après la lecture du mot ab ? Reprendre la question avec les mots $\varepsilon, a, aba^2b, a^2ba^2b, ab^4$ et b^3a^2. Parmi ces mots, lesquels sont reconnus par A_1 ? Lesquels sont des blocages ?

3. Décrire les mots reconnus par l'automate A_1.

4. L'automate A_2 est il émondé ? Déterministe ? Complet ?

5. Donner un exemple de mot de longueur 4 reconnu par A_2 et un non reconnu par A_2. Donner une expression rationnelle simple dénotant le langage L_2 reconnu par A_2.

6. Proposer un automate reconnaissant le complémentaire de L_2. Justifier votre construction.

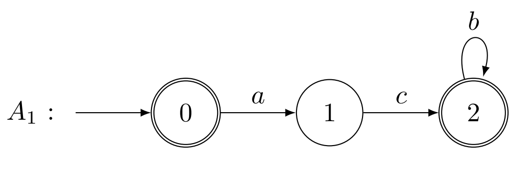
Exercice 2 (*) *Dessine moi un automate*

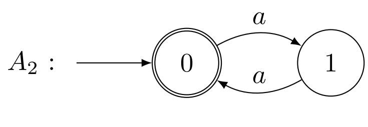
Pour chacun des langages L suivants sur l'alphabet $\{a, b\}$, dessiner un automate **déterministe** qui le reconnaît. On ne demande pas de preuve formelle du fait que l'automate proposé reconnaît effectivement le langage voulu mais la construction proposée doit être suffisamment claire et explicable pour qu'on en soit convaincu.

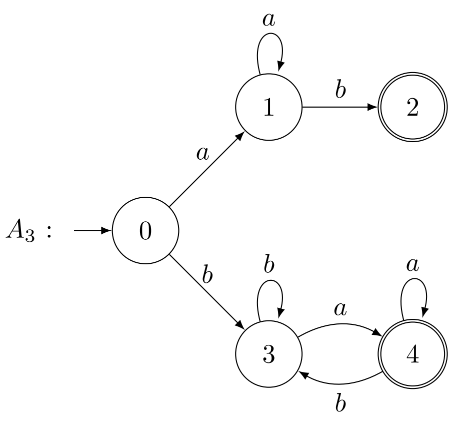
1. $L = \{a, \varepsilon\}$
2. $L = \{maa \mid m \in \{a, b\}^\star\}$
3. $L = \{a^m b^n a^p \mid m, n, p \in \mathbb{N}\}$
4. L est dénoté par $(a^2)^\star$
5. L est dénoté par $(a + b)^\star aba(a + b)^\star$
6. L est le langage des mots qui contiennent au plus un a
7. L est le langage des mots admettant aba comme sous-mot
8. L est le langage des mots qui ne contiennent pas abb
9. (**) En expliquant la démarche, contruire un automate déterministe sur $\{a, b, c\}$ qui reconnaît le langage des mots qui contiennent soit le motif $ab^\star a$, soit le motif $b^\star cb^\star$ mais pas les deux.

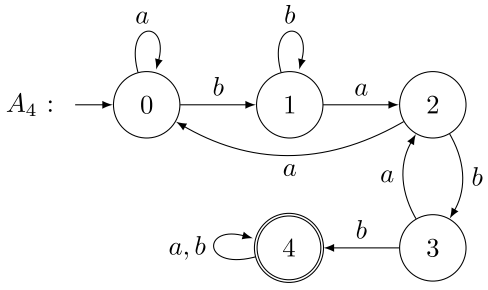
Exercice 3 (*) *Langages reconnus*

Pour chacun des automates ci-dessous, donner en expliquant (mais sans nécessairement donner de preuve formelle) une expression régulière décrivant le langage qu'il reconnaît.









Exercice 4 (**) *Reconnaissance de multiples*

Nous avons étudié en cours la construction d'un automate A_3 reconnaissant les écritures binaires d'entiers divisibles par 3 (lesdites écritures étant potentiellement non purgées de 0 en tête).

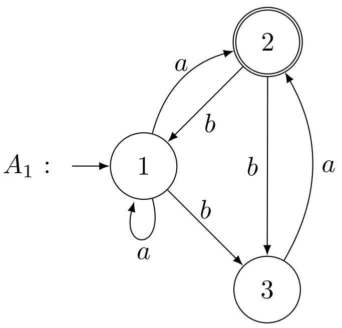
1. Proposer un automate déterministe et complet A_2 reconnaissant les mots sur $\{0, 1\}$ qui sont écriture binaire (potentiellement non purgée) d'entiers pairs.

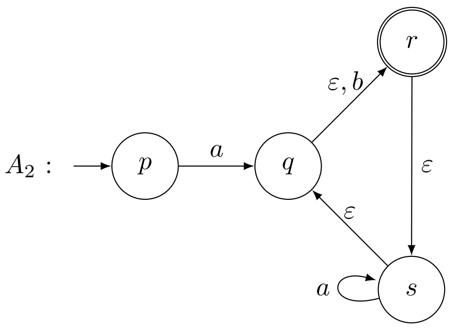
2. Calculer l'automate produit de A_2 et A_3. Quel langage reconnaît-il ?

3. En s'inspirant des techniques vues en cours et ci-dessus, déterminer un automate reconnaissant les écritures binaires d'entiers divisibles par 10.

Exercice 5 (*) *Déterminisations*

1. Déterminiser les deux automates ci-dessous. Quel langage reconnait A_2 ?





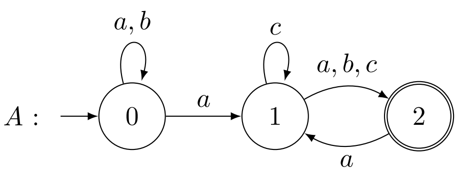
2. Proposer un automate non déterministe permettant de reconnaître le langage sur $\{a, b\}$ des mots dont l'avant dernière lettre est un b. Déterminiser l'automate obtenu.

3. Reprendre la question précédente avec le langage des mots dont $aaba$ est suffixe.

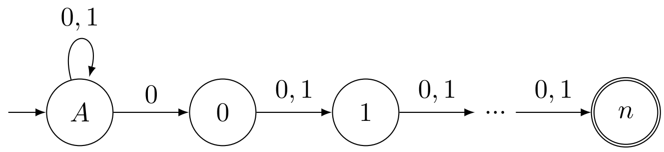
4. L'algorithme de déterminisation exposé en cours construit nécessairement un automate dont tous les états sont accessibles. Les états dudit automate sont-ils nécessairement tous coaccessibles ? Justifier.

Exercice 6 (**) *Une déterminisation coûteuse*

1. Déterminiser l'automate suivant. Quel est le nombre d'états du déterminisé ? Commenter.



L'observation faite ci-dessus est en fait générale : pour tout $n \in \mathbb{N}$ il existe un automate à n états dont la déterminisation produit un automate ayant un nombre d'états exponentiel en n. Pire : il existe un automate A à n états tel que le plus petit automate déterministe (en termes de nombre d'états) reconnaissant le même langage que A a un nombre d'états exponentiel en n. Considérons l'automate A_n suivant :



2. Donner une expression rationnelle pour le langage reconnu par A_n.

3. Déterminiser A_1 puis A_2.

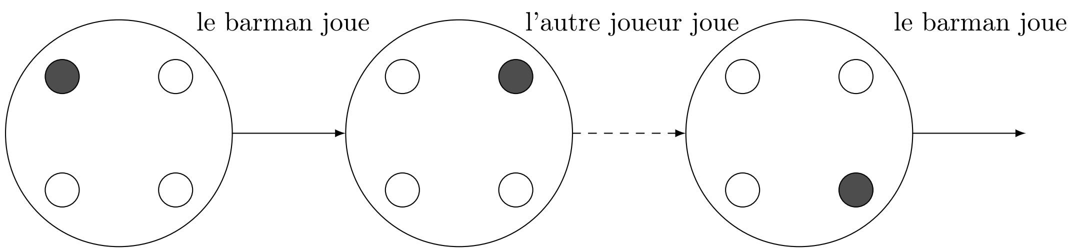
4. Combien d'états semble avoir le déterminisé de A_n ?

5. (***) Montrer qu'aucun automate déterministe possédant strictement moins de 2^n états ne reconnaît $L(A_n)$. *Remarque : Ceci montre que le déterminisé qu'on obtient génériquement en appliquant les méthodes de la question 3 est minimal.*

Exercice 7 (***) *Le barman aveugle*

Deux joueurs font face à un plateau tournant sur lequel sont placés 4 jetons aux coins d'un carré. Ces jetons ont deux faces de couleurs différentes : l'une est blanche, l'autre est noire. L'un des deux joueurs (le barman) a les yeux bandés. Son objectif est de faire en sorte que les 4 jetons soient retournés sur la même couleur : dès que cela arrive, il gagne. Pour ce faire il peut retourner $1, 2$ ou 3 jetons à chaque tour. Entre chaque tour, l'autre joueur fait pivoter le plateau d'un quart de tour, d'un demi tour ou de trois-quarts de tour.

On peut ainsi avoir la suite de coups suivants : le barman joue en retournant les deux jetons du haut sans les voir puis l'autre joueur fait pivoter le plateau d'un quart de tour. C'est maintenant au barman de jouer :



1. Expliquer pourquoi il n'y a que quatre configurations du plateau fondamentalement différentes. En s'aidant de cette observation, modéliser le jeu à l'aide d'un automate.

2. Montrer que le barman a une stratégie gagnante, c'est-à-dire une suite de coups qui le fera gagner quoi que fasse le joueur qui tourne le plateau.

Exercice 8 (**) *Suppression des ε-transitions*

Nous avons vu en cours que l'algorithme de déterminisation présenté dans le cas d'automates sans ε-transitions permet de déterminiser également les automates à ε-transitions. C'est un moyen de supprimer les transitions instantanées mais il est parfois très coûteux (voir exercice 6). Dans cet exercice on montre qu'on peut supprimer les ε-transitions sans pour autant déterminiser.

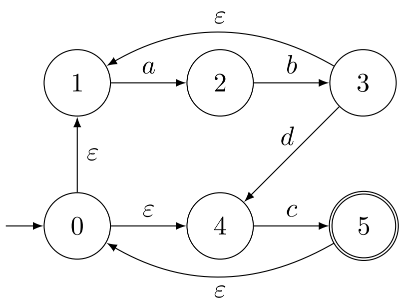
On rappelle que la clôture instantanée d'un ensemble est l'union des clôtures instantanées de ses éléments et que $\delta(E, a)$ où E est un ensemble d'états est l'union des $\delta(e, a)$ pour $e \in E$. Soit $A = (\Sigma, Q, I, F, \delta)$ un automate à ε-transitions. On définit l'automate A' par $A' = (\Sigma, Q, I', F, \eta)$ où I' est la clôture instantanée de I et pour tout $q \in Q$ et tout $a \in \Sigma$, $\eta(q, a)$ est la clôture instantanée de $\delta(q, a)$.

1. Prouver par récurrence sur $|u|$ que pour tout $u \in \Sigma^\star$, $\delta^\star(I, u) = \eta^\star(I', u)$.

2. En déduire que A' est un automate sans ε-transition reconnaissant le même langage que A.

Remarque : On vient de prouver que tout AFND-ε est équivalent à un AFND ayant le même nombre d'états.

3. A l'aide de la méthode présentée ci-dessus, supprimer les ε-transitions de l'automate suivant.



4. Quel est le coût de la suppression des ε-transitions sur un automate à n états via cette méthode ? Le comparer au coût au pire cas d'une déterminisation et conclure.

Exercice 9 (***) *Algorithme de minimisation de Moore*

Si $A = (\Sigma, Q, \{q_0\}, F, \delta)$ est un automate déterministe complet, on définit pour tout $q \in Q$ le langage $L_q = \{u \in \Sigma^\star \mid \delta^\star(q,u) \in F\}$ puis la relation d'équivalence suivante sur Q :

$$q \sim q' \text{ si et seulement si } L_q = L_{q'}$$

On note $\overline{q}$ la classe d'équivalence de q selon cette relation. On admet dans cet exercice que l'automate $A' = (\Sigma, Q', \{q_0'\}, F', \delta')$ tel que :

- $Q' = \{\overline{q} \mid q \in Q\}$
- $F' = \{\overline{q} \mid q \in F\}$
- $q_0' = \overline{q_0}$
- pour tout $\overline{q} \in Q'$ et toute lettre $a \in \Sigma$, $\delta'(\overline{q}, a) = \overline{\delta(q,a)}$.

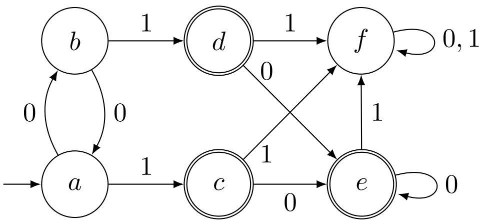
a un nombre d'états minimal parmi tous les automates déterministes et complets reconnaissant $L(A)$ et on l'appelle l'automate minimal reconnaissant $L(A)$. On calcule les classes d'équivalences selon $\sim$ en calculant successivement les classes d'équivalence selon $\sim_k$ telles que

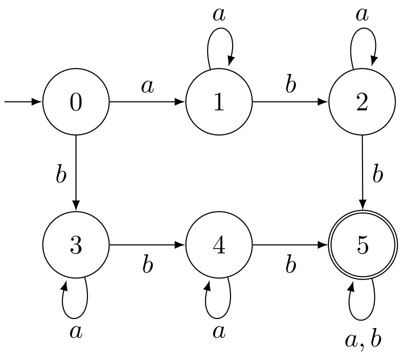
$$q \sim_k q' \text{ si et seulement si } \{u \in L_q \mid |u| \leq k\} = \{u \in L_{q'} \mid u \leq k\}$$

1. Déterminer les classes d'équivalences selon $\sim_0$.
2. Montrer que pour tous $q, q' \in Q$ et tout $k \in \mathbb{N}$: $q \sim_{k+1} q' \Leftrightarrow (q \sim_k q'$ et $\forall a \in \Sigma,\ \delta(q,a) \sim_k \delta(q',a))$.
3. Montrer qu'à partir d'un certain rang $k \in \mathbb{N}$, $\sim_k$ coïncide avec $\sim$.

L'algorithme sous-jacent aux questions précédentes est l'algorithme de Moore. Mettons-le en application :

4. Dans les deux cas suivants déterminer les classes d'équivalence d'états de l'automate A selon $\sim$ et en déduire un automate minimal reconnaissant le langage $L(A)$:





Remarque : Un DM prochain fera explorer plus avant le concept et les techniques de minimisation d'automates.